Project Specification

Project Name: ATLAS Second Level Trigger Prototype RoI Builder

Version 2.0
Dec. 16, 2002

1. Scope

This project encompasses production and testing of a prototype system for interfacing the ATLAS first level trigger to the second level trigger.

2.Related projects and documents

1.ATLAS High Level Triggers, DAQ and DCS Technical Proposal at http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/SG/TP/tp_doc.html

2.S-link documentation at

3.Specification of the LVL1 / LVL2 trigger interface at https://edms.cern.ch/document/107485/1

4.A Prototype ROI Builder for the Second Level Trigger of ATLAS Implemented in FPGA's, R.Blair et al., LEB'99, Snowmass, September 20-24 1999.

5.RoIB Requirements at http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/DataFlow/DataCollection/docs/DC-014.pdf

6.The level-1/level-2 interface: RoI Unit, Y.Ermoline, ATLAS DAQ note 94-34, 8 December 1994.

7.The Level 2 Supervisor Requirements, at http://press.web.cern.ch/Atlas/GROUPS/DAQTRIG/DataFlow/DataCollection/docs/DC-009.pdf

8.Technical Manual, 8101/8104 Gigabit Ethernet Controller, LSI Logic, November, 2001

9.Altera Data Book, Altera Corporation, San Jose, California

3.Technical Aspects

The Region of Interest Builder (RoIB) is intended to build records from data received from the level 1 trigger elements, select a target supervisor processor, and distribute the records at high input rate to a number of commodity PC's. Figure 1 shows a use-case indicating how the RoIB interacts with the first level and high level trigger (HLT). The RoIB takes raw event fragments from various level one sources and accumulates all the fragments of a given event and then sends the complete event information to one of a number of supervisor PC's. A single PC will receive all of the data from a given event and from there the data will be distributed to HLT systems that require it for further event selection and disposition. Using this *divide and conquer* approach a single PC never sees the full level 1 rate and it can easily manage the required IO.

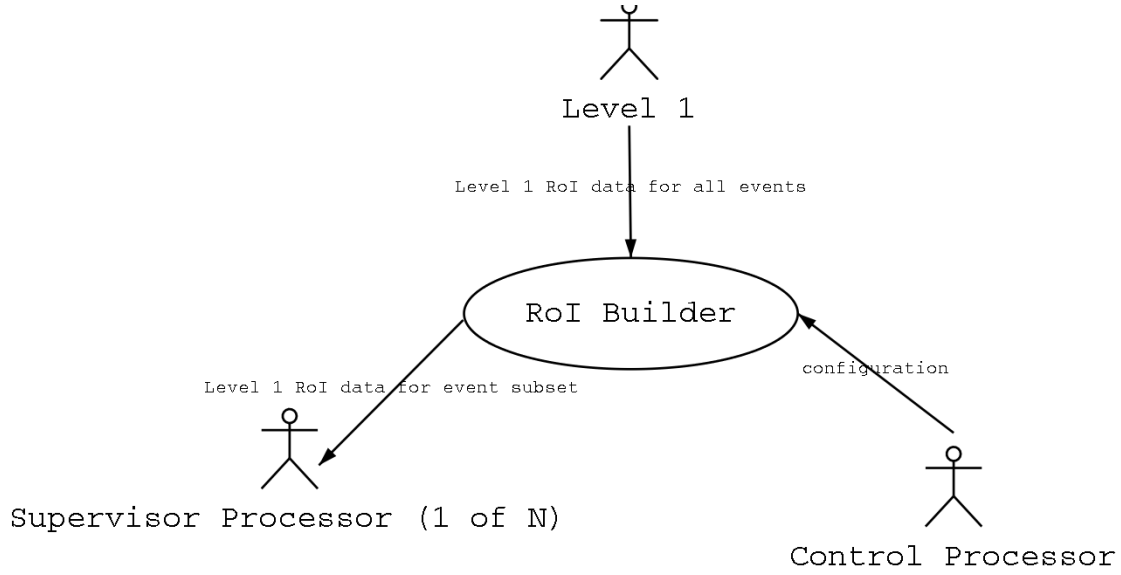

Figure 1: Use-case showing the RoI Builder context.

3.1 Requirements and Specifications

The RoIB must satisfy the following requirements

- Collect data from the level 1 sources
- Assemble event data
- Send a subset of event data to each target supervisor processor
- Interface with a control system to allow for run coordination and reset functions
- Operate at level 1 event rate
- Handle corrupted or missing input data in a well defined manner that allows for proper error recovery

The prototype RoIB will satisfy all of the requirements of the final system, but may be limited to fewer supervisor processors and fewer level 1 data sources than the final RoIB.

3.2 Technical Description

The architecture is conceptually similar to the prototype RoI Builder that was developed for the hardware integrations of trigger elements that were accomplished over the last few years. This prototype RoI Builder was described in our paper for the real time conference at Snowmass in 1999. Basically, the system implemented a highly parallel architecture realized in FPGA's. Incoming fragments were distributed to several record building channels. Using the embedded Level 1 ID's the logic was able to allocate RoI fragments from particular events to the correct channels.

In this new prototype RoI Builder the RoI fragments will be brought to the RoI builder via S-link. Each fragment contains data collected from a portion of the level 1 trigger system. The level 1 information required for the level 2 system is the collection of all such fragments for an event. This includes both the information about the trigger decision as well as eta and phi data for the subsystems that cause an event trigger. We will refer to the collected RoI fragments for a given event as an RoI record. The RoI Builder (RoIB) input card will pass fragments to a set of builder cards. Each builder card communicates RoI records to up to four supervisor processors. We plan to transfer the compiled RoI records to the target supervisor processors using S-link (see figure 2). Each of the builder cards is responsible for a subset of the events that trigger level 1.

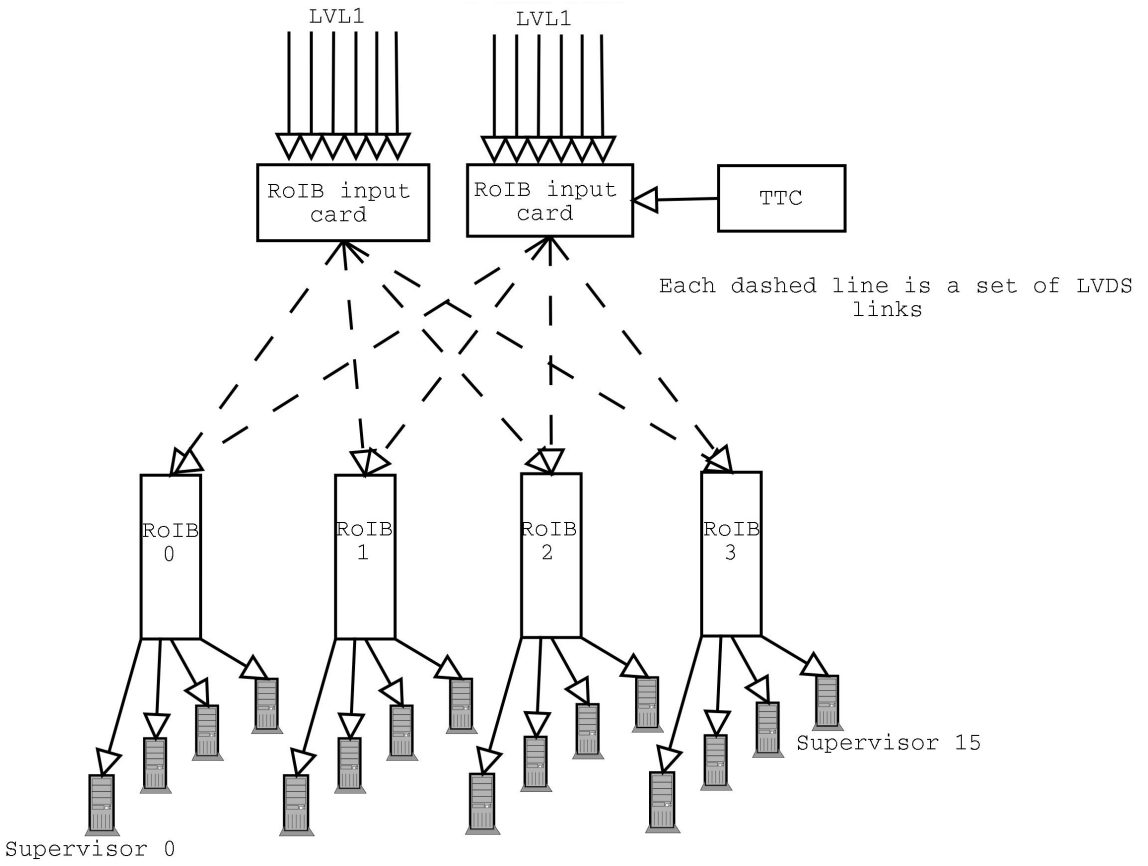

Figure 2: Diagram showing the RoIB and how the level 1 data is expected to be distributed.

In the original prototype the system selected the target supervisor processor strictly by a round robin algorithm. In this new prototype RoI Builder there will be several algorithms, and we intend to make them considerably more sophisticated. In all cases the system will skip target processors where flow control is active. The system is expandable in units of four supervisor processors by adding another RoI Builder card. The system will be able to accommodate up to four RoI Builder cards.

We intend that the RoI builder will implement a number of event selection algorithms, which can be selected and configured from VME. Any algorithm must treat flow control properly and must deal with timeouts. Several algorithms have been suggested, as follows:

1.The events are allocated to supervisor processors on a round robin basis, with the hardware dealing automatically with the number of cards, etc. If a channel is asserting flow control, that channel is simply skipped. The algorithm ignores level 1 ID.

2.Each RoI builder card is more or less autonomous. Events are allocated to RoI builder cards on the basis of Mod(ID, # of cards). Where flow control is asserted the channel is skipped and the event allocated to the next channel on the card.

3.The first card handles all events unless busy in which case the events are allocated to the next card.

4. Some quality of the events other than event ID could be used as a parameter to allocate the events to particular cards and consequently particular supervisor processors.

In every case the timeout system must interact with the event selection algorithm so that if a fraction is missing the problem is handled properly. We expect to make the event selection system very flexible so that algorithms may be implemented in the future, or may be modified as desired. A simple round robin selection scheme is likely adequate but flexibility could prove useful. As an example ATLAS is currently considering multiplexing schemes for the readout system that would slice the system by event ID's (use different hardware for subsets of events). In this case algorithm 2 might allow for a better coordination between "slices" of readout and "slices" of the level 2 farm.

It is essential that the system be able to function in the presence of flow control. It is not easy to build records arriving from a multiplicity of sources when flow control is going on and off from various elements, but it is important that the data integrity not be affected. We plan to have deep FIFO's on every input so that the peaks of data activity will be averaged, but will have flow control going back to the individual S-link source cards. In order to assure this we have worked through in detail how the system will respond we focus on the case where the algorithm is the first in the above list. An interaction diagram (since this is hardware it is actually only an “interaction like” diagram) shows the decision tree exercised by the system in this particular case (see figure 3).

The individual RoI fractions can be as long as 63 S-link words including headers and trailers, and are in the S-link format. It is necessary to accommodate the time skew of arriving RoI fragments, and accordingly a timer is started at the arrival of the first fragment of each event. If all the fragments have been received before the timeout the compiled record will be transferred to the target supervisor processor. If the timeout occurs first the system executes a course of action which has been selected via VME. This course of action could be to discard the incomplete record or it could be to build the record from the incomplete set of fragments and forward it with an error flag to the target processor. The timeout and other parameters are of course selectable from VME. The maximum value of timeout that the system can implement is a critical parameter. To the extent that a partially built RoI record has to wait for fragments the ROI builder must provide buffering so that other records can be built concurrently. The RoI Builder will accommodate a timeout as long as 1ms.

3.2.1 Additional architectural details

The RoIB is composed of a number of 9U RoI Builder cards which receive RoI Fragment information in the form of S-link from as many as 12 Level 1 Trigger elements, and provide RoI Records in standard S-Link format through J3 to Transition cards as S-Link to as many as 4 Supervisor Processors. In addition to the ability to communicate via S-Link, the ROI Builder cards will communicate with each other via auxiliary card jumpers.

The ROIB will utilize 9U RoI Input cards, which will reside in the RoIB crate. RoI Input cards have 6 standard S-Link connectors which receive standard S-Link Link Destination Cards so that any physical medium may be used to import RoI Fragments from Level 1, and provide fan-out in LVDS transceivers. When a Fragment is received in standard S-Link format on an Input Card, the fragment is first written to a FIFO 4K words deep to provide time buffering. When a FIFO is non-empty and there is a free RoI builder card to receive it, it will be distributed to the builder cards via LVDS connections at the back of the VME crate. RoI Input cards also have on-board diagnostic memories which may be initialized from VME, and used to provide diagnostic data for execution of diagnostic software. An RoI Builder consisting of 2 RoI Input cards and 4 RoI Builder cards can receive input fragments from 12 Level 1 Trigger elements and provide RoI Records to 16 Supervisor Processors.

3.2.1.1 Event Allocation

It had been envisioned that several allocation algorithms would be implemented in the RoIB, with the algorithm which allocated the Supervisor Processor selected under software control. Although we intend to implement several algorithms, for the present discussion we will describe only one implementation in detail. The algorithm is not implemented in one location, but instead is implemented in both the input and builder cards. There are two steps involved in the allocation. The first is the selection of the card where the record is built, and the second is the choice of assembly unit (AU) on the card. There is a hierarchy to the cards and a hierarchy to the AUs on each card. The cards pass a token from one to another, and only the card with the token may build the next event. When a card receives the token, it allocates the next event to the appropriate AU. If an AU is busy waiting for a tardy fragment, the card allocates the next event to the next AU in the hierarchy. When a card allocates the next event to its last AU, the token is passed to the next card. If, when the token is passed to a card where all the AUs are busy, the token is immediately passed to the next card.

When a card receives the token, it will try to allocate the next event to the AU at the top of the hierarchy on that card. Since fragments will continue to be received for events being built on the previous card, it is essential that each card maintain a log of recent event ID's of event fragments that arrived while it did not have the token, say for example the last 16. In this way the card with the token can identify the new events in the presence of tardy fragments from previously allocated events.

The use of a VME based system is motivated by data sharing between components. The RoIB will use a high density connector on J3 similar to magic bus. The data rates for the RoIB are not particularly high, but there need to be a number of data paths in and out. Each RoIB card needs 12 data paths in (assuming we accommodate up to 12 level one trigger elements), and 4 data paths out for the 4 supervisor processors supported by one RoIB card. The card will be implemented in about 6 20k200e FPGA's which could conceivably fit on a 6U card, but the difficulty of squeezing the design down to a 6U card is unjustified. The system will be designed for deployment in a 9U crate.

In any allocation algorithm it is important that if a Supervisor Processor crashes or is otherwise unable to accept RoI Records, the channel corresponding to that Supervisor Processor is skipped by the algorithm and the record passed to the next channel. The RoI Builder knows that the Supervisor channel is unavailable if Flow Control is active for that link or if the link is down. Accordingly, when the allocation algorithm selects an AU, and hence a Supervisor Channel for an event, it is essential that the algorithm pass over channels that cannot process records.

3.2.1.2 Flow Control

Flow Control can become active at a number of points in the RoIB/Supervisor system, and it is important that it be taken into account properly. Flow Control should be effective in dealing with situations where the event rate temporarily exceeds the maximum average rate that the system can accommodate. Where the event rate exceeds the maximum average rate the RoIB can process for long periods of time the Flow Control system and corresponding back pressure will have to force trigger deadtime thus losing events.

In the implementation of the RoIB-Supervisor link there is a FIFO 4K words deep at the input of the Link Source Card (LSC). Flow Control at this input is raised by the logical OR of the Almost Full of this FIFO and the Watermark of the Transmit FIFO in the MAC. This watermark can be set very low so that Flow Control will always be raised immediately when an ROI Record is sent to a particular LSC. The logic always requires that when Flow Control is raised during the transmission of a fragment or record the whole fragment or record be transmitted, and then nothing further be transmitted until Flow Control is removed. Accordingly, if an ROI Record is allocated to a supervisor no other ROI Record will be allocated until the first is transmitted, and if that Supervisor has crashed or that link is otherwise incapable of transmitting the record, that link will remain inactive. In this way event loss can be limited to at worst a single event if a Supervisor processor crashes. The data for this event should be available to VME for a more complete recovery.

Within each builder card in the RoIB there are input FIFO's 4K words deep on each of the 12 input data streams. These FIFO's receive the incoming ROI Fragments from the input cards, and provide buffering. Input shift register buffers follow the FIFO's, and if a FIFO is non-empty and the following shift register buffer is not occupied, the fragment at the top of the FIFO is strobed into the shift register buffer. At this time the allocation algorithm determines if the fragment is an element of a record to be built on this card. If so, the algorithm determines which of the four assembly units on the card should receive the fragment, and it is shifted to the appropriate assembly unit.
In the event that the allocation algorithm determines that the fragment is not an element of a record to be built on this card, the fragment is discarded and the next fragment shifted from the input FIFO if non-empty.

If an AU has been selected by the allocation algorithm, it collects fragments with that

EVENTID until it has received a fragment from each active input. If a subset of these fragments is delayed the AU must wait. While the AU is waiting there is no need to raise Flow Control because this AU will not be selected by the allocation algorithm. After some period if a subset of fragments is still missing the AU times out and the incomplete record must be flagged and transmitted to the Supervisor Processor. If the tardy fragment or fragments are subsequently received they must be discarded, so the logic must be aware that this record has already been built.

Since Flow Control is never raised by a busy AU the input shift register buffer need only shift in a fragment whenever the FIFO it follows is non-empty. The problem arises when the allocation algorithm has allocated events to all the AUs and they are all waiting for a tardy subset of fragments and have not yet timed out. The allocation algorithm must be able to ascertain that all the AUs on all builder cards are busy. At that time the new fragments received by the RoIB must remain in the input FIFO's, and the FIFO's for the tardy channels will be empty. If the Level 1 Accepts happen at a rate of 100 KHz then the FIFO's can accommodate something like 100 events before becoming full. This implies that the system can wait approximately 1 millisecond for the tardy subset of fragments. This also says something about the allowable variation in the latencies of the various level 1 trigger elements. Of course the input FIFO's could be twice as deep and double the allowable amount of time. In any event, when an input FIFO is almost full it must raise Flow Control, and send Flow Control back through S-Link to the Level 1 element. It is important that the ROIB logic requires records to be built such that all fragments have identical Level 1 Event ID's. For example, if the RoIB waits for a tardy fragment, times out, and subsequently sends the incomplete record to the Supervisor Processor, and then the tardy fragment arrives, the allocation algorithm must recognize that this fragment is not an element of a record to be built, but must be treated specially.

Flow Control can be raised by a Supervisor processor or by the link to it. This will be sensed by the allocation algorithm and that AU will be passed over for subsequent events. If the allocation algorithm cannot allocate an event to a builder (and hence a Supervisor Processor), it must suspend operation until an output is available. If supervisor Processors are available, but complete records cannot be built, the builders must wait for the tardy fragments. Timers can be started when the first fragment of an event is allocated to a builder, and a time can be set for the builder to time out and build an incomplete record, which can be tagged and sent on to the appropriate Supervisor Processor. If some subset of fragments is tardy and all builders are waiting to build records, the input FIFO's will begin to fill. When they are almost full they will raise flow control on the links to the Level 1 Trigger elements.

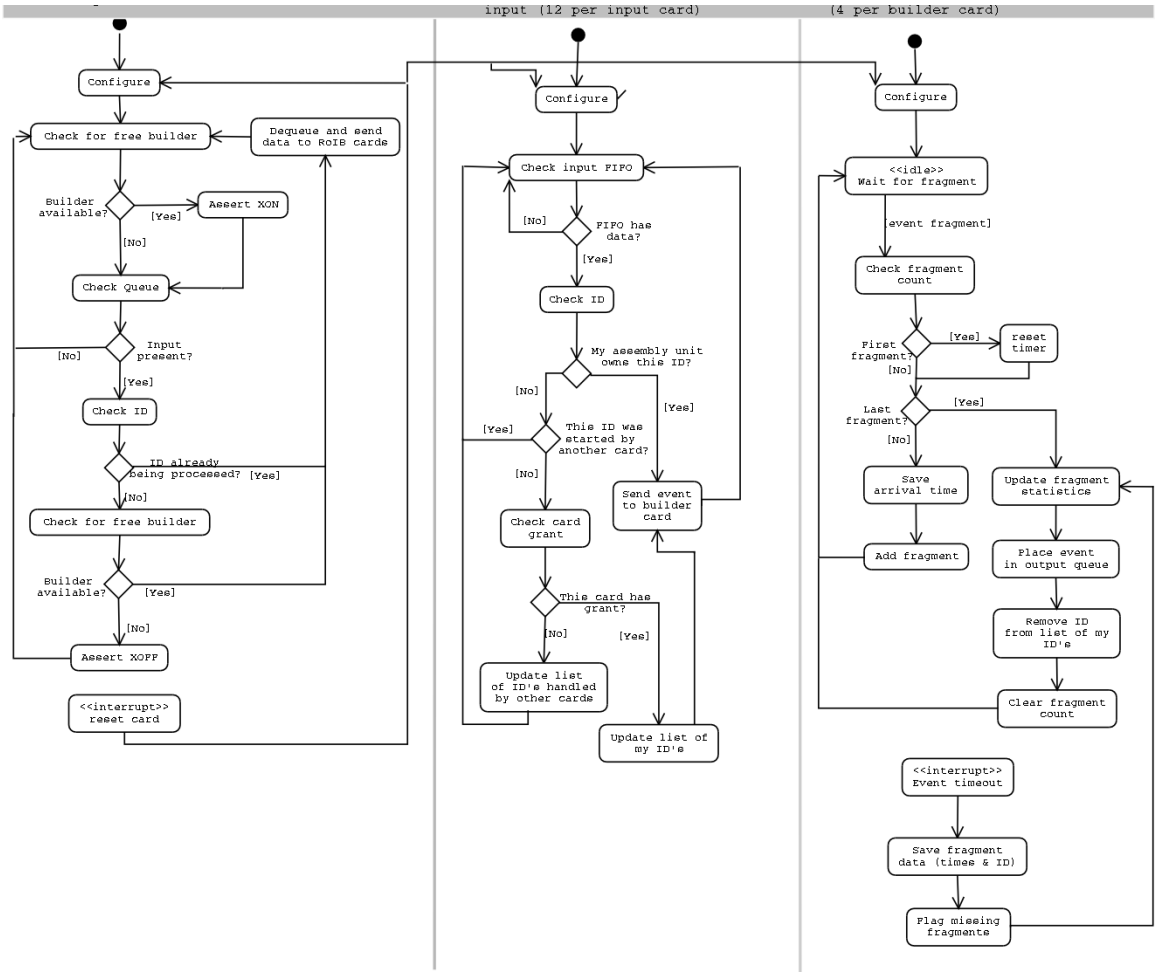

Figure 3: Diagram indicating the logic and flow control aspects of the RoIB system given a simple round robin style selection.

3.2.1.3 Error Handling

A number of exceptional conditions may arise while running and the system needs to be able to gracefully handle them and also provide useful feedback on what hardware if any is failing when they occur. Normal flow control, described above, will smooth out traffic bursts, but if the average Level 1 rate exceeds the system capacity flow control will exert back pressure on the Level 1 RODs and this in turn should throttle Level 1.
The link to the Level 1 subsystems is standard S-Link and error handling on this link should be the same as that used throughout the rest of the experiment in the ROD to ROB interface. The link from the input cards to the builder cards is LVDS and can be regarded as redundant. The normal redundancy provided by a link per fragment makes recovery possible since the Level 1 ID of the event is carried by the fragments that are sent normally. For full event recovery the DAQ can use this to retrieve the missing fragment since it is stored in a ROB on an independent data path.

The link to the supervisors will be busy until the supervisor has read the next event (i.e. the FIFO is only used to hold a single output event). If data is corrupted on this link the supervisor should initiate a recovery procedure. Here the Level 1 ID is not carried on redundant links so the RoIB will have to provide the pending output Level 1 ID via VME to the crate controller when the DAQ recovers the event. This can be achieved by having a register that can be interrogated for each output link. When an error is detected the supervisor will have to assert XOFF until the event ID is recovered via this VME

recovery procedure.

The logic will be implemented in Altera 10K FPGA's, using a 3.3 Volt technology or perhaps a 2.5 Volt technology. The logic will be very dense and will be designed such that there is a large amount of resources unused in the FPGA's which allows for changes, modifications, and additional features to be implemented easily.

Grounding considerations should not be significant since all input and output is via fiber Gigabit Ethernet and the cards will be designed as standard 9U VME boards.

4.Monitoring

There are a number of quantities that should be monitored in order to anticipate errors and to evaluate the causes of malfunctions as well as to act as a cross check on other system components. An exhaustive list is not yet available but some items that should be available as histograms and as values that can be retrieved for each AU in the system will include:

1. fragment arrival times (. the initial fragment) for each input
2. input corresponding to the first fragment for an event
3. input corresponding to an out of order fragment (i.e. a fragment that is not the same ID as all other inputs for the Nth event)
4. input corresponding to any fragments with BCIDs different from the other fragments of an event
5. AU event counts
6. AU current event and preceding 15 events
7. flow control state/FIFO size for all queues in the system

Another cross check and a feature that would allow for more faithful simulation of collider running is the addition of a TTC input as a 13th Level 1 component. The RoI Builder will include an additional TTC input that will act as an addition Level 1 component. The TTC input can be used to emulate Level 1 by providing an input to the RoIB when Level 1 is not available. The TTC input can be used to verify that the Level 1 ID/ trigger was properly sent to readout components via TTC before the Supervisor sends the event to Level 2. This allows yet another crosscheck on the BCID corresponding to the trigger**.**

3.3 Manufacturing

Argonne plus outside suppliers, as needed, will provide the PCB's and component assembly.

3.4 Testing

The testing will be done in several stages. Communications tests will be performed at Argonne using existing PCs and adapters with software modified from that used to test the Gigabit Link Source Card. After rudimentary checks on the functionality binary tests will be scheduled with several level 1 systems followed by tests with at least two such systems simultaneously. These tests will be similar to those performed with the previous

pre-prototype RoIB. Unlike the previous tests the control functions for the RoIB will be integrated into the software framework and will conform to the current online framework for level 2. Final tests should be performed with a vertical slice of the trigger including representative pieces of both the level 2 and level 1 trigger.

3.5 Installation

The system will consist of one or more 9U VME cards. Fiber connectors will be on the front panel. LEDs on the front panel will indicate media connection and activity.

3.6 Maintenance and Further Orders

This system is intended as a prototype and will not be in service for more than four years. Sufficient cards will be produced initially to accommodate any need for maintenance. Any future production will involve significant modification based on evaluation of the prototype performance and other desirable improvements resulting from advances in related technology.

4.Project Management

Currently the funding and project management are coordinated by the US ATLAS Project office at Brookhaven National Laboratory. Periodic reviews and monthly reports are coordinated by the project management team. The current US ATLAS TDAQ level 2 manager is R. Blair.

4.1 Personnel

Institution
Extension

Customer
R. Blair
ANL
X7545

Project Engineer
J. Dawson
ANL
X7525

Software Professional
J. Schlereth
ANL
X6281

4.2 Milestones and Schedule

Test milestones are yet to be determined. This needs to be done in collaboration with the level 1 group. An initial PDR meeting occurred in Feb. 2002. Initial card design and

fabrication should be complete before March 2003. Schematic level review will be done by Feb. 2003.

4.3 Costs and Reviews

All manufacturing, assembly and component costs will come under WBS 1.6 of the US ATLAS project management plan. Monthly progress will be reported via the US ATLAS reporting system. There will be a PDR signoff prior to final design and a FDR prior to production.

4.4 Safety

General laboratory safety codes apply.

1.Environmental Impact

4.5.1 Disposal

ANL will dispose of cards at the end of their life.

4.5.2 EMC

Since the modules are prototypes they will be outside the scope of the EMC regulations. However, since the electronics must function as designed, without malfunction or unacceptable degradation of performance due to electromagnetic interference (EMI) within their intended operational environment, the electronics shall comply with specifications intended to ensure electromagnetic compatibility.

4.6 Handling Precautions

Anti-static precautions bust be taken when handling the card to prevent damage to expensive components.